# Paper Presentations

Divij Khaitan

July 12, 2024

# Contents

Analysis of Trends in Maximum and Minimum Temperature

Temperature Prediction using Machine Learning

Temperature Prediction using Fuzzy Time Series

# Summary

- Data from CRU, 1901-2002 at $1 \times 1$ resolution
- No trends found from 1901-1930, so analysis starts from 1931
- Increasing linear trends for max and min temperatures in the South
- No significant trend in DTR
- Negative linear relationship between DTR and cloud cover

# Methods

- Using the modified bartlett test, the paper finds no significant changes in variance across the different grids over the 70 years i.e. all the grids showed similar values for variance
- Only look at summer (July, August) and winter (January, February)
- Cloud cover and temperature were modelled linearly with respect to DRT and year respectively
- Serial correlation was low, but significance was tested with and without it
- Spatial patterns were found using spatial kriging, which is the following model for a location $s$

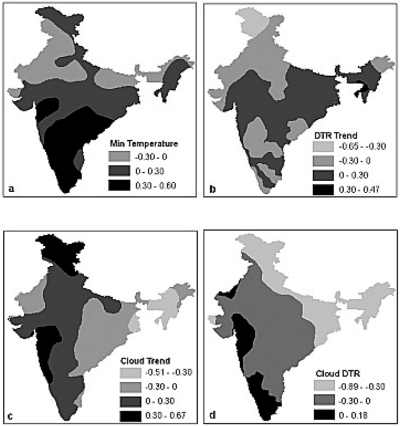
$$Z(s) = \mu + \varepsilon(s)$$

# Results: Summer

- Increasing trends in minimum temperature, decreases seen in the North and Northwest
- Trend in cloud cover showed an east-west divide, increasing in the East apart from the Thar Desert
- DTR increasing in Central and Northeast, decreasing in the North and mixed in the South
- Positive relationship of cloud cover and DTR along patches on the West coast, but negative everywhere else

# Visualisation: Summer

Figure: Summer results for trends in tmin, DTR, cloud cover, DTR-cloud cover relationship



## Results: Winter

- Increase in minimum temperature in the South, Northeast and Kashmir
- No 'significant' trend in DTR
- Cloud cover declining all over the country, most sharply in the Northwest and the Southeast
- Positive relationship of cloud cover and DTR in the South, but negative everywhere else

# Visualisation: Winter

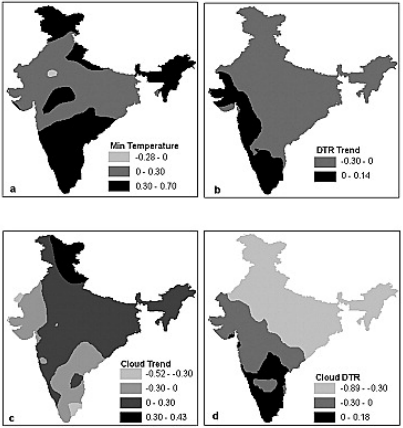



Figure: Winter results for trends in tmin, DTR, cloud cover, DTR-cloud cover relationship

# Summary

- Predicting $t_{max_i}$ using $t_{max_{i-1}} \dots t_{max_{i-k}}$
- SVMs outperform feedforward networks
- Performance of SVMs stagnates after $k = 5$ days, MLP performance degrades after 5 days

# Methods

Given some dataset $\{(x_i, y_i)\}_{i=1}^n$, where $x_i$ is a vector of the maximum temperature on each of the previous days

- Support Vector Machines

$$\begin{array}{ll} \underset{w,b}{\text{minimize}} & \frac{1}{2}\|w\|^2 \\ \text{subject to} & |y_i - (w \cdot x_i + b)| \leq \epsilon \quad \forall i. \end{array}$$

- Neural networks

$$\underset{\theta}{\text{minimize}} \quad \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}(f(x_i;\theta), y_i),$$

Where $f(x_i;\theta) = A_n(\phi(A_{n_1}(\phi \ldots (\phi(A_1 x) \ldots)))),$
$\theta = (\text{vec}(A_1), \ldots, \text{vec}(A_n))$

# Results

- Tested on a single dataset of weather data from Cambridge from 2003 - 2007
- Replaced missing values with monthly means, standardised input to SVM but normalised input to MLP
- SVM outperforms MLP at every context length
- MLP performance begins degrading with context windows greater than 5, and SVM stagnates beyond 5 days

# Fuzzy Set Theory

- A fuzzy set on a universal set $U$ is $\{(u, \mu_A(u)) \mid u \in U\}$, where $\mu_A : U \to [0, 1]$ represents the degree of membership of $u$ in set $A$
- A fuzzy time series is a $F(t) : \mathbb{Z}^+ \to \mu(t)$ that maps a point in time to a fuzzy set
- A autoregressive time series can be represented using the max-min composition operation $\circ$

$$F(t) = F(t-1) \circ R(t, t-1)$$

- Assume $F(t-1) = \{(0.5, x_1), (0.8, x_2)\}$ and $\begin{bmatrix} 0.7 & 0.4 \\ 0.6 & 0.5 \end{bmatrix}$ is the relationship matrix, then $x_1$ can be computed as $x_1(t) = x_1(t-1) \circ R(t, t-1) =$ $\max(\min(0.5, 0.7), \min(0.8, 0.6)) = 0.6$

# Fuzzy Set Theory

- When $R$ is constant, the series is called time invariant. Otherwise, it is called time variant
- When predicting $F(t)$ from $G(t)$, they are called the main and second factors respectively

# Converting data into a fuzzy set

1. Divide the observation space into $n$ disjoint intervals $(u_1, \ldots, u_n)$
2. Define $k$ fuzzy sets $A_i = \{(u_j, \mu_{A_i}(u_j)) | j \in [n]\}$
3. Let $x = F(t) - F(t-1)$. If $x \in u_i$, then we need to find the set $A_k$ such that $\mu_{A_k}(x) \geq \mu_{A_{k'}}(x) \forall k \neq k'$
4. The fuzzified variation between $t$ and $t-1$ is defined to be $f(t) = A_k$

For the main time series the differences between sucessive observations are considered, but for the secondary factor just the raw values are mapped to bins and converted into fuzzy sets

## Operation Vectors Matrices and Factor Vectors

- The criterion vector is defined as

  $$C(t) = f(t-1) = [\mu_{A_k}(u_1)\ \mu_{A_k}(u_2)\ \ldots\ \mu_{A_k}(u_n)]$$

  The operation matrix is denoted by
  $O^w(t) = [C(t-1)^T C(t-2)^T \ldots C(t-w)^T]^T$, where
  $O_{ij} = C(t-i-1)_j = f(t-i)_J$
- The factor vector is defined with respect to the secondary factor as

  $$S(t) = g(t-1) = [S_1\ S_2\ \ldots\ S_n]$$

# Defining the Relation

- For a single time series, the relationship can be captured using the criterion vector and operation matrix
- For a 2 factor prediction problem, we also need to incorporate the factor vector, using the following equation

$$R_{ij} = O_{ij} \times S_j \times C_j$$

- When combined with the update rule described earlier, $F(t) = F(t-1) \circ R(t, t-1)$, we get the update rule

$$f(t) = [\max_i(R_{i1}), \max_i(R_{i2}) \ldots, \max_i(R_{in})]$$

- The data used is from Taiwan in 1996, for the months June to September

# Algorithm B

1. Partition the data into groups and perform the algorithm independently on each group. The split chosen here was month-wise
2. Find the largest increase $D_R$ and largest decrease $D_L$, and fix the range of the main time series to be $U = [D_L - D_1, D_R + D_2]$
3. Partition the data into even-length intervals $u_1, \ldots, u_n$
4. Define fuzzy sets on $U$. This paper considers 7 intervals and 7 fuzzy sets, organised like a tridiagonal matrix which is 1 on the diagonal and 0.5 on the super and sub diagonals

# Algorithm B

5. The secondary factor incorporates human knowledge. It uses the fact that cloud density is inversely proportional to mean temperature, and the fact that extremely high/low cloud cover is likely to drop/increase the next day, the matrix for this looks like

$$\begin{bmatrix} 0 & 0 & 0.5 & 1 & 1 & 1 & 1 \\ 0 & 0.5 & 1 & 1 & 1 & 1 & 1 \\ 0.5 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0.5 \\ 1 & 1 & 1 & 1 & 1 & 0.5 & 0 \\ 1 & 1 & 1 & 1 & 0.5 & 0 & 0 \end{bmatrix}$$

6. Fuzzify both time series

# Algorithm B

7. Compute $O^w, C, S$ and use these to compute $R$
8. Compute $f(t)$ using the update rule
9. If all the membership in all the intervals is 0, forecast no change from the previous values. Else, look at the intervals with the largest membership values, $u_{(1)}, \ldots, u_{(k)}$ and forecast the arithmetic means of their midpoints $\frac{\sum m_{(i)}}{k}$

An alternate algorithm B* is proposed which only considers forecasts if the maximum membership value is above some threshold $\alpha$, else forecasts 0. Further, the maximum and minimum values are also computed for the data, and any forecasts greater/lesser than the upper/lower bounds are clipped to them

# Results

- Slight performance improvement from Algorithm A (baseline one-factor fuzzy forecasting)
- B and B* have similar performance
- General trend of % error increasing with an increase in window size $w$

The algorithm is efficient, but the dataset was extremely small and this may need a lot of parameter tuning before results are meaningful